

Minimal surface computation using a finite element method on an embedded surface

Mirza Cenanovic^a, Peter Hansbo^a, Mats G. Larson^b

^a*Department of Mechanical Engineering,
Jönköping University*

SE-55111 Jönköping, Sweden

^b*Department of Mathematics and Mathematical Statistics
Umeå University*

SE-901 87 Umeå, Sweden

Abstract

We suggest a finite element method for computing minimal surfaces based on computing a discrete Laplace–Beltrami operator operating on the coordinates of the surface. The surface is a discrete representation of the zero level set of a distance function using linear tetrahedral finite elements, and the finite element discretization is done on the piecewise planar isosurface using the shape functions from the background three dimensional mesh used to represent the distance function. A recently suggested stabilization scheme is a crucial component in the method.

1 Introduction

An important application of partial differential equations on surfaces is equations that are used to determine the shape of surfaces in order to satisfy certain design criteria, called *form finding*, cf. [1]. A classical example is the minimal surface problem, the simplest kind of form finding, where a surface with minimal curvature is sought, given the position of its boundary.

In this paper, we consider minimizing curvature by means of discretizing the Laplace-Beltrami operator on an embedded surface, following Olshanskii, Reusken, and Grande [7], where a Galerkin method is constructed by using the restrictions of continuous piecewise linears defined on embedding tetrahedra to the embedded surface. The Laplace-Beltrami operator applied to the Cartesian coordinates of the surface gives the curvature vector $(\kappa_1 + \kappa_2)\mathbf{n}$, where κ_1 and κ_2 are the principal curvatures and \mathbf{n} is the surface normal, cf. [1]. Our algorithm is motivated by previous work on meshed surfaces by Dziuk [3], where viscous relaxation was used to move a triangulated surface on which the Laplace-Beltrami operator was discretized. A method on embedded surfaces related to ours is given by Chopp [2], where the curvature is computed using instead the values of a level set implicitly defining the embedded surface. Classically, this problem has also been considered in a referential domain, cf. Johnson and Thomée [6], which severely limits the scope of the method.

2 Model problem and finite element method

2.1 The continuous problem

Let $\Gamma(t)$ denote a time-dependent smooth two-dimensional surface embedded in \mathbb{R}^3 with signed distance function ϱ . For notational convenience, we shall frequently omit the dependence on t in $\Gamma(t)$.

We consider the following problem: given a final time T , find $\mathbf{x}_\Gamma : \Gamma(t) \rightarrow \mathbb{R}^3$ such that

$$\dot{\mathbf{x}}_\Gamma - \Delta_{\Gamma(t)}\mathbf{x}_\Gamma = 0 \quad \text{in } \Gamma(t), \quad t \in (0, T), \quad (1)$$

$\mathbf{x}_\Gamma = \mathbf{g}$ on $\partial\Gamma(t)$ (we shall assume \mathbf{g} is constant). Here

$$\dot{\mathbf{x}}_\Gamma := \frac{\partial \mathbf{x}_\Gamma}{\partial t},$$

Δ_Γ is the Laplace-Beltrami operator defined by

$$\Delta_\Gamma = \nabla_\Gamma \cdot \nabla_\Gamma \quad (2)$$

where ∇_Γ is the surface gradient

$$\nabla_\Gamma = \mathbf{P}_\Gamma \nabla \quad (3)$$

with $\mathbf{P}_\Gamma = \mathbf{P}_\Gamma(\mathbf{x})$ the projection of \mathbb{R}^3 onto the tangent plane of Γ at $\mathbf{x} \in \Gamma$, defined by

$$\mathbf{P}_\Gamma = \mathbf{I} - \mathbf{n} \otimes \mathbf{n} \quad (4)$$

where $\mathbf{n} = \nabla \varrho$ denotes the exterior normal to Γ at \mathbf{x} , \mathbf{I} is the identity matrix.

As is well known, cf. [3],

$$-\Delta_{\Gamma(t)} \mathbf{x}_\Gamma = 2H\mathbf{n}$$

where $H = (\kappa_1 + \kappa_2)/2$ is the mean curvature of $\Gamma(t)$. Thus, we have the curvature driven normal flow

$$\dot{\mathbf{x}}_\Gamma = -2H\mathbf{n}.$$

If \mathbf{x}_Γ is found by following the zero isosurface of a level set function $\phi(\mathbf{x}, t)$, then the material derivative of the level set function at \mathbf{x}_Γ is given by

$$\frac{d\phi}{dt} = \frac{\partial \phi}{\partial t} + \dot{\mathbf{x}}_\Gamma \cdot \nabla \phi = 0, \quad (5)$$

and, since $\nabla \phi = \mathbf{n}$, if we assume that $|\nabla \phi| = 1$ is enforced (i.e., the level set function is a distance function), we may update the level set function at the surface directly via

$$\frac{\partial \phi}{\partial t} = 2H.$$

This idea, together with a relation between H and spatial derivatives of ϕ , was used by Chopp [2]. Here we shall instead use (1) directly.

Defining

$$V = \{\mathbf{v} \in [H^1(\Gamma)]^3 : \mathbf{v} = \mathbf{0} \text{ on } \partial\Gamma\},$$

the weak statement corresponding to (1) takes the form: given the coordinate map or embedding of $\Gamma = \Gamma(t)$ into \mathbb{R}^3 , denoted by $\mathbf{x}_\Gamma : \Gamma \ni \mathbf{x} \mapsto \mathbf{x} \in \mathbb{R}^3$, find the velocity $\dot{\mathbf{x}}_\Gamma =: \mathbf{u}_\Gamma \in V$ such that

$$(\mathbf{u}_\Gamma, \mathbf{v})_{\Gamma(t)} + a(\mathbf{x}_\Gamma, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in V \quad (6)$$

where

$$a(\mathbf{u}, \mathbf{v}) = (\nabla_\Gamma \mathbf{u}, \nabla_\Gamma \mathbf{v})_{\Gamma(t)}, \quad (7)$$

and $(\mathbf{v}, \mathbf{w})_\Gamma = \int_\Gamma \mathbf{v} \cdot \mathbf{w} \, d\Gamma$ is the L^2 inner product on Γ .

2.2 Discretization in time and space

In Dziuk [3], a semi-discrete version of (6) of backward Euler type on meshed surfaces, yielding a discrete velocity \mathbf{u}_Γ^h , so that given the nodal coordinates \mathbf{x}_N^n , $\mathbf{x}_N^n \approx \mathbf{x}_N(t_n)$, \mathbf{x}_N^{n+1} was computed by

$$\mathbf{x}_N^{n+1} = \mathbf{x}_N^n + k_n \mathbf{u}_\Gamma(\mathbf{x}_N^n), \quad (8)$$

where $k_n = t_{n+1} - t_n$, updating the mesh, updating n , and continuing until the curvature is small enough. We now wish to instead solve this problem on embedded surfaces in \mathbb{R}^3 using the general technique proposed by Olshanskii et al. [7] by means of a distance function for the definition of Γ and we cannot thus directly solve for the location of Γ but need to solve the additional equation (5) for the level set function.

To discretize in space, let \mathcal{K} be a quasi uniform partition into shape regular tetrahedra of a domain Ω in \mathbb{R}^3 completely containing Γ^n for all n . Furthermore, we let

$$\mathcal{K}_h^n = \{K \in \mathcal{K} : K \cap \Gamma^n \neq \emptyset\} \quad (9)$$

be the set of tetrahedra that intersect Γ^n . See Figure 1.

We let

$$V_h^n = \{\mathbf{v} : \mathbf{v} \text{ is a continuous piecewise linear polynomial defined on } \mathcal{K}_h^n, \mathbf{v} = \mathbf{0} \text{ on } \partial\Gamma^n\}.$$

In general, a finite element method for computing the mean curvature vector by means of the discrete Laplacian may fail due to instability, see [5] where the following stabilization method was suggested: given Γ^n and the corresponding coordinate function \mathbf{x}_Γ^n , find $\mathbf{u}_\Gamma^h \in V_h^n$ such that

$$(\mathbf{u}_\Gamma^h, \mathbf{v})_{\Gamma^n} + j(\mathbf{u}_\Gamma^h, \mathbf{v}) = -a(\mathbf{x}_\Gamma^n, \mathbf{v})_{\Gamma^n} \quad \forall \mathbf{v} \in V_h^n \quad (10)$$

where the bilinear form $j(\cdot, \cdot)$ is defined by

$$j(\mathbf{u}, \mathbf{v}) = \sum_{F \in \mathcal{F}_I} ([\mathbf{t}_F \cdot \nabla \mathbf{u}], [\mathbf{t}_F \cdot \nabla \mathbf{v}])_F. \quad (11)$$

Here, \mathcal{F}_I denotes the set of internal interfaces in \mathcal{K}_h^n , $[\mathbf{t}_F \cdot \nabla v] = (\mathbf{t}_F \cdot \nabla v)^+ - (\mathbf{t}_F \cdot \nabla v)^-$ with $w(\mathbf{x})^\pm = \lim_{s \rightarrow 0^+} w(\mathbf{x} \mp s\mathbf{t}_F)$, is the jump in the tangent gradient across the face F . Here, the jump in the tangent gradient at an edge E shared by the elements K_1 and K_2 is defined by

$$[\mathbf{t}_E \cdot \nabla \mathbf{u}] = \mathbf{t}_{E,K_1} \cdot \nabla \mathbf{u}_1 + \mathbf{t}_{E,K_2} \cdot \nabla \mathbf{u}_2 \quad (12)$$

where $\mathbf{u}_i = \mathbf{u}|_{K_i}$, $i = 1, 2$, and \mathbf{t}_{E,K_i} denotes the outwards unit vector orthogonal to E , tangent to K_i , $i = 1, 2$. See figure (2) This stabilization method was shown to yield first order convergence of the curvature in $L_2(\Gamma)$ in the finite element method proposed in [5].

After obtaining \mathbf{u}_Γ^h we use (5) to get the level set evolution equation

$$\frac{\partial \phi}{\partial t} = \mathbf{u}_\Gamma^h \cdot \mathbf{n} = 0, \quad (13)$$

where we employ a time discretization scheme to solve for the level set function at time t_{n+1} , ϕ^{n+1} .

Note that this method requires that the level set function is a distance function, $|\nabla \phi|=1$. This is done by reinitialization of the level set function. In order to keep computational effort at a minimum both the reinitialization and the propagation are being done on a narrow band of tetrahedral elements so that only a small set of elements that are cut by the surface and their neighbor elements are updated.

3 Numerical implementation

Below follows a step by step finite element implementation.

1. Construct a linear tetrahedral mesh \mathcal{K} is created on the domain Ω in \mathbb{R}^3 in which we embed the the implicit surface Γ . Let \mathbf{x}_N denote the vector of node coordinates in \mathcal{K} .
2. Set up the level set function $\phi(\mathbf{x}, t_n)$ such that $\phi(\mathbf{x}_\Gamma, t_n) = 0$.
3. Discretize the distance function $\phi^{h,n}(\mathbf{x}) \approx \phi(\mathbf{x}, t_n)$ by evaluating it in the nodes of the tetrahedral mesh giving a nodal vector $\boldsymbol{\phi}_N^n = \phi(\mathbf{x}_N, t_n)$.
4. Initialize $\mathbf{x}_N^0 = \mathbf{x}_N(0)$
5. Find elements that are cut by the surface, \mathcal{K}_h^n , using (9).
6. Extract zero isosurface points giving Γ_h^n by going over all elements \mathcal{K}_h^n interpolating the signed distance function linearly using the tetrahedral basis functions.

7. Compute the velocity field $\mathbf{u}_{\Gamma^n}^h$ by solving (10). This is done by solving the matrix equation

$$\mathbf{M} \mathbf{u}_N = -\mathbf{S} \mathbf{x}_N^n \quad (14)$$

where \mathbf{u}_N are the nodal velocities in the band of elements containing Γ^n , \mathbf{S} is the matrix corresponding to the Laplace–Beltrami operator, and \mathbf{M} a stabilized mass matrix computed on Γ^n .

8. Choose a time step k_n and propagate $\phi^{h,n}$ to $\phi^{h,n+1}$, in the nodes of the band of elements containing Γ^n , using

$$\phi^{h,n+1} = \phi^{h,n} - k_n \mathbf{n} \cdot \mathbf{u}_{\Gamma^n}^h \quad (15)$$

where

$$\mathbf{n} = \frac{\nabla \phi^{h,n}}{|\nabla \phi^{h,n}|} \quad (16)$$

is the normal vector at \mathbf{x} (in practice we also use an L_2 projection to represent \mathbf{n} at the nodes).

9. Reinitialize $\phi^{h,n+1}$: we reinitialize the function near the front by computing on a narrow band of elements using the time step as a guide for how many elements to select. The reinitialization can be done in several ways see, e.g., [8]; we chose to reinitialize by updating the value of $\phi^{h,n+1}$ in each node in the narrow band with the signed distance to the closest node on the discrete surface Γ_h^{n+1} .
10. Compute the discrete mean curvature

$$H^h := -\frac{\mathbf{n} \cdot \mathbf{u}_{\Gamma^n}^h}{2},$$

which should converge to zero everywhere.

11. If the L_2 norm $\|H^h\|_{\Gamma_h} \leq \epsilon$, where ϵ is a small number, then stop; otherwise update $n \mapsto n + 1$ and repeat from step 5.

4 Numerical examples

The following Figures show some examples of minimal surfaces computed using the scheme of Section 3. We give some examples on different surface-evolutions. In all convergence plots the curvature has been computed as the Euclidean norm of $H^h(\mathbf{x}_N)$ (mean curvature evaluated in the nodes).

Fig 3 shows a cylinder evolving to a catenoid. The initial radius is 0.5, with axis centered at $x = 0$, $y = 0$, the dimensions are taken from [2]; the height is 0.554518 and the mesh domain is of the size $1.2 \times 1.2 \times 0.554518$ and consists of 14761 tetrahedral elements. Figure 3 shows the evolutions from a cylinder to a catenoid with a inner diameter of approximately 0.4 which agrees with the theoretical result. The face colors represent a velocity field where dark is high velocity. Figure 5 shows the convergence of mean curvature for the catenoid.

In Fig 5 a cut cylinder is shown. The initial radius is 0.5, with axis positioned at $x = 0$, $y = 0.06$. The height is 0.554518 and the mesh domain is of the size $1.2 \times 1.2 \times 0.554518$, there are 23040 tetrahedral elements. Figure 6 shows the evolution from a cut cylinder to a cut catenoid. Figure 6 shows the convergence of mean curvature for the cut catenoid.

Fig 7 shows a cylinder evolving into two flat circles. The initial radius is 0.5 and the axis is positioned at $x = 0$, $y = 0$. The height is 1 and the mesh domain is of the size $2 \times 2 \times 1$. There are 47017 tetrahedral elements. Figure 8 shows the convergence of mean curvature for the collapsing cylinder.

Figs 9–11 shows an evolving Schwarz minimal surface starting from a sphere of radius 0.5. The mesh used 196608 tetrahedra of which approximately 24976 were active at any given time. See Figure 12 for the convergence of the curvature.

5 Concluding remarks

We have proposed a novel way of computing minimal surfaces using a discretization of the Laplace–Beltrami operator on a 2D surface embedded in a 3D mesh. The approach has a strong theoretical foundation in terms of proven accuracy of the computed curvature vector used to drive the evolution of the surface. In future work, we will consider more complex surface energies, e.g., leading to membrane elasticity on the surface as in the recent work by Hansbo and Larson [4].

References

- [1] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy. *Polygon Mesh Processing*. A K Peters, Ltd., Natick, MA, 2010.

- [2] D. L. Chopp. Computing minimal surfaces via level set curvature flow. *J. Comput. Phys.*, 106(1):77–91, 1993.
- [3] G. Dziuk. An algorithm for evolutionary surfaces. *Numer. Math.*, 58(6):603–611, 1991.
- [4] P. Hansbo and M. G. Larson. Finite element modeling of a linear membrane shell problem using tangential differential calculus. *Comput. Methods Appl. Mech. Engrg.*, 270:1–14, 2014.
- [5] P. Hansbo, M. G. Larson, and S. Zahedi. Stabilized discrete Laplacians and computation of mean curvature of triangulated surfaces. Preprint 2013.
- [6] C. Johnson and V. Thomée. Error estimates for a finite element approximation of a minimal surface. *Math. Comp.*, 29:343–349, 1975.
- [7] M. A. Olshanskii, A. Reusken, and J. Grande. A finite element method for elliptic equations on surfaces. *SIAM J. Numer. Anal.*, 47(5):3339–3358, 2009.
- [8] J. A. Sethian. *Level set methods and fast marching methods*. Cambridge University Press, Cambridge, 1999.

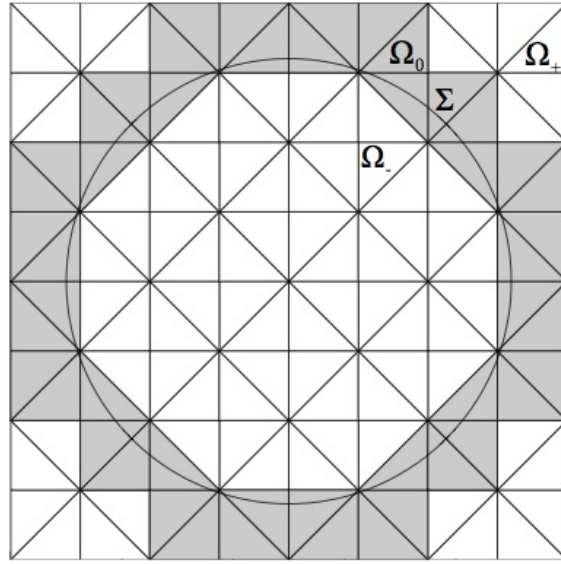


Figure 1: Intersecting domains

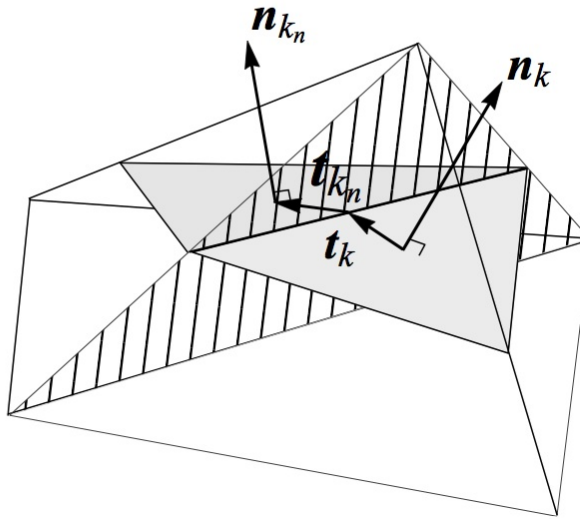


Figure 2: Tangential jump

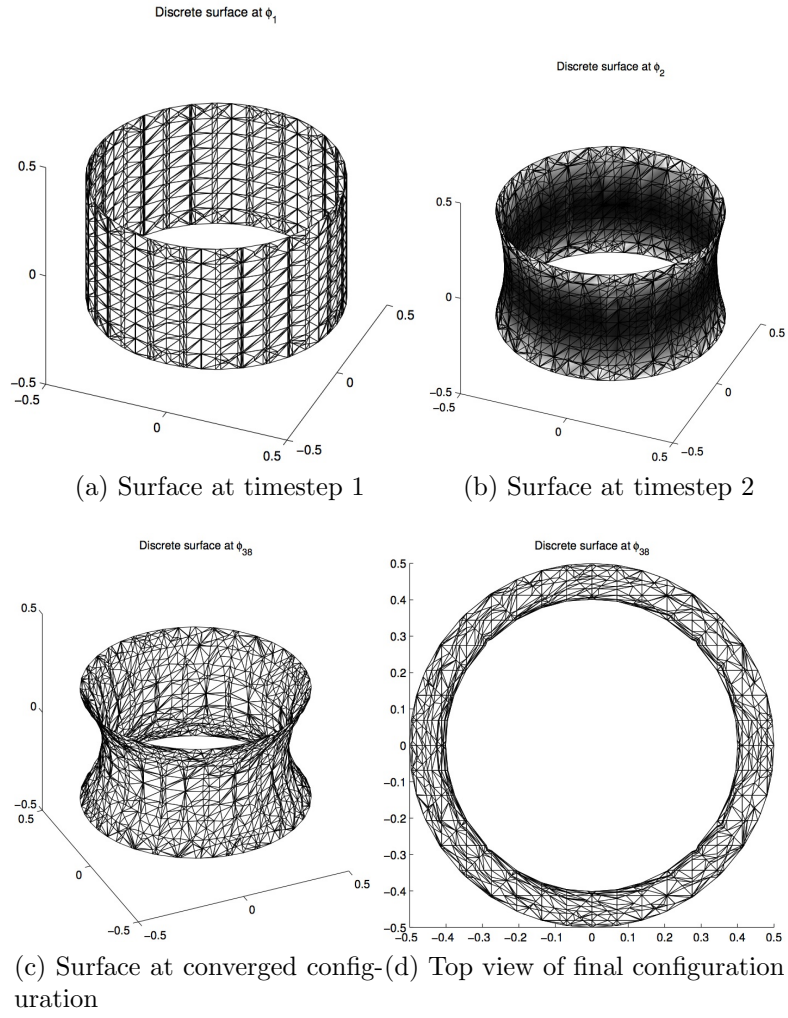


Figure 3: Catenoid

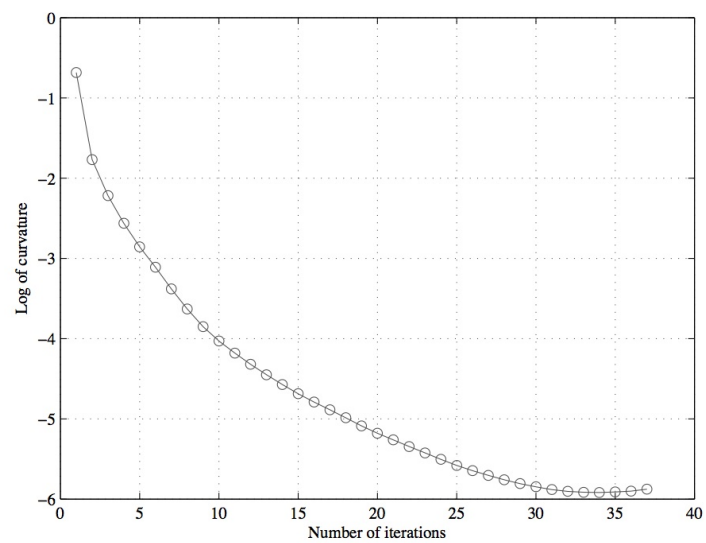


Figure 4: Convergence of the curvature

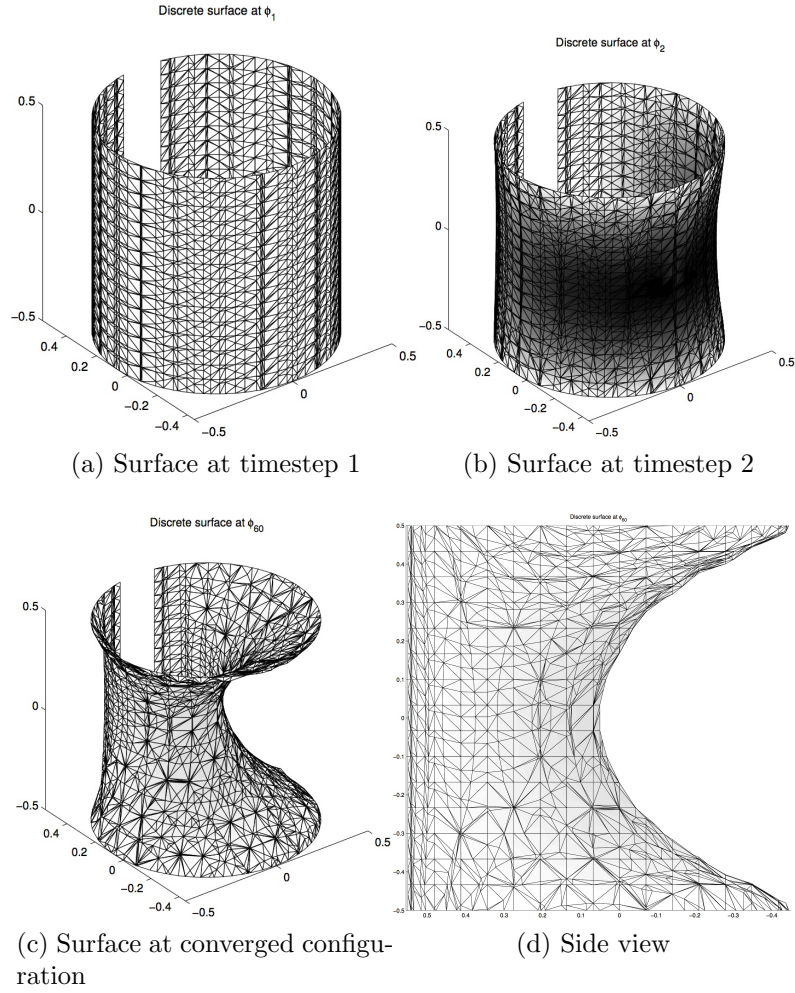


Figure 5: Cut catenoid

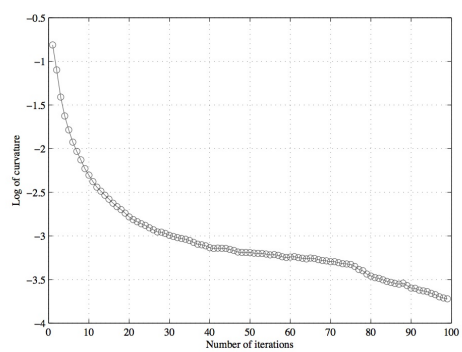


Figure 6: Convergence of the cut catenoid curvature

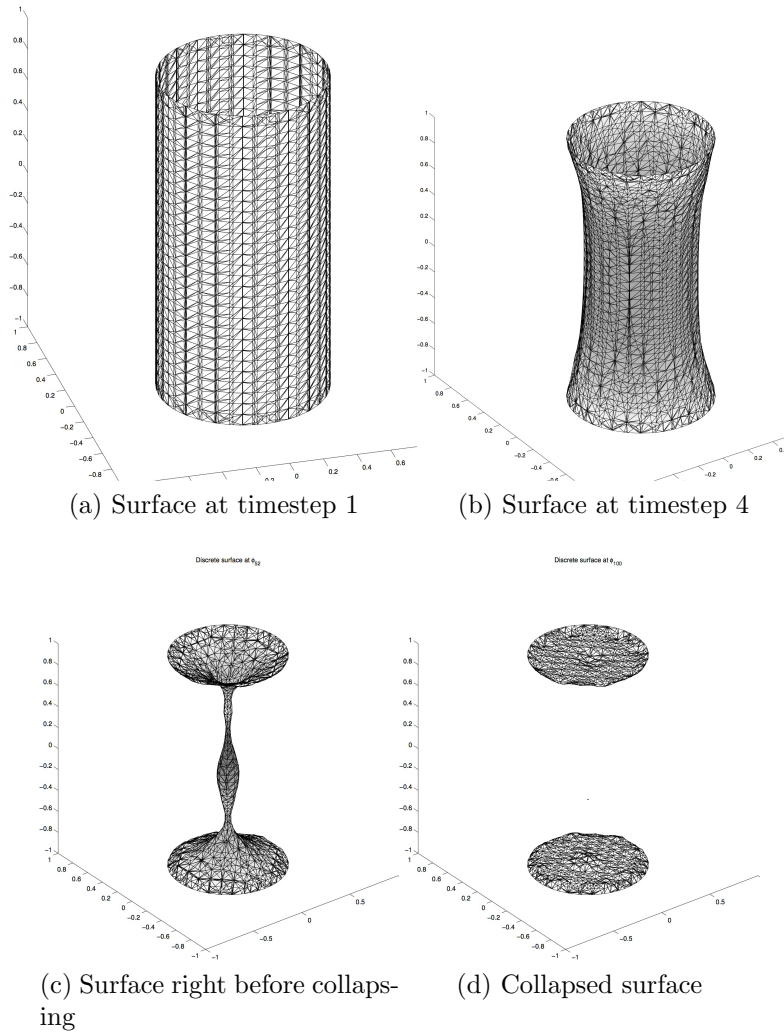


Figure 7: Cut catenoid

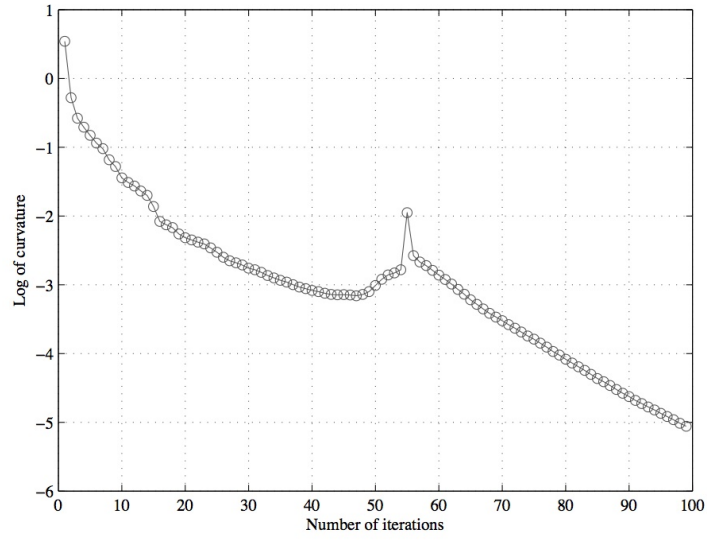


Figure 8: Convergence of the collapsing cylinder curvature

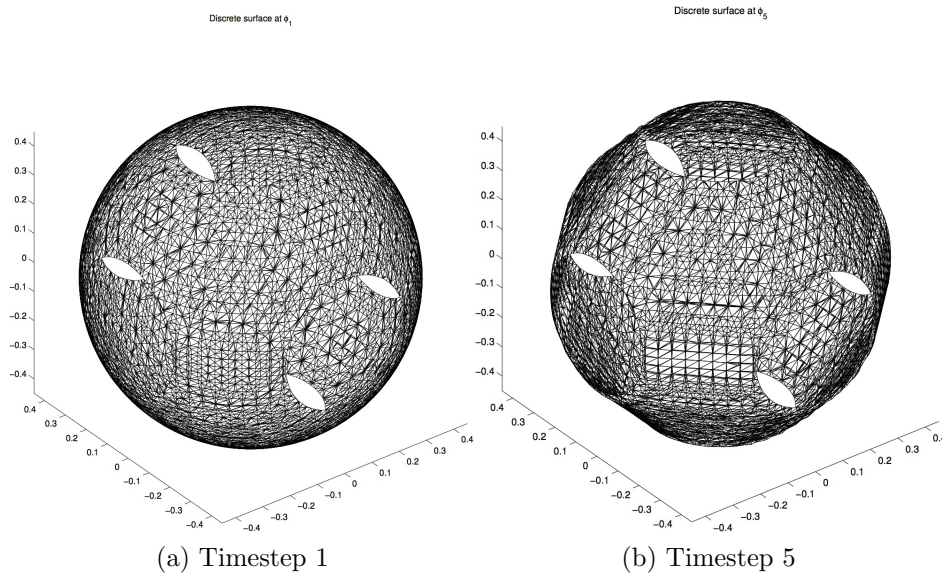


Figure 9: Schwarz minimal surface, early time

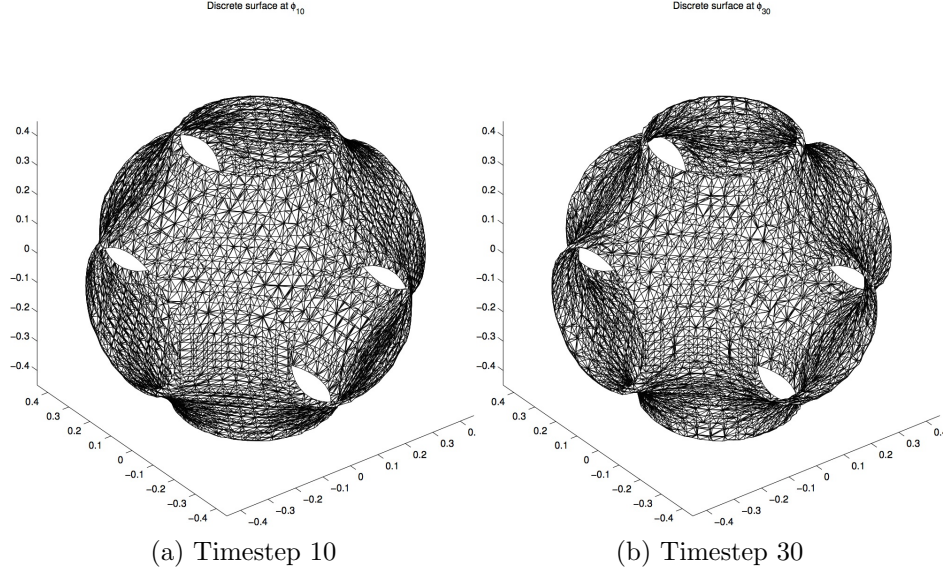


Figure 10: Schwarz minimal surface, intermediate time

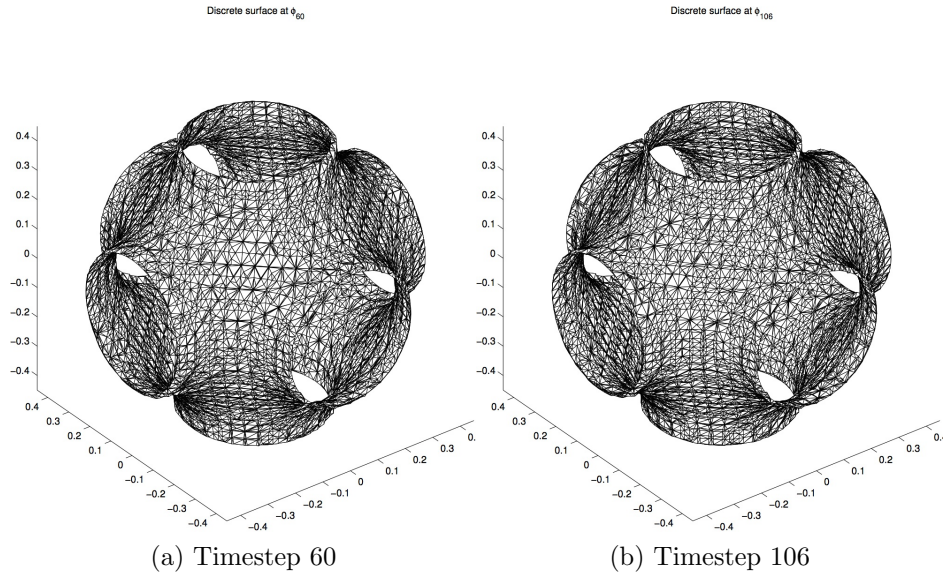


Figure 11: Schwarz minimal surface, late time

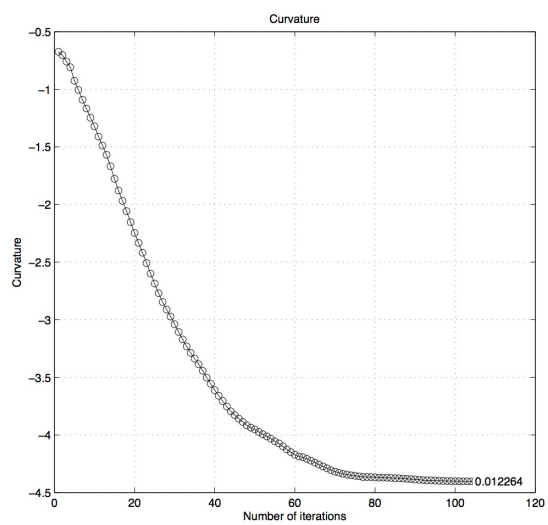


Figure 12: Convergence of the Schwarz surface